END
DATE
FILMED
1 -80
DDC

MICROCOPY RESOLUTION TEST CHART

NATIONAL BUREAU OF STANDARDS-1963-A

LEVEL

RADC-TR-79-240, Vol I (of three)
Final Technical Report
October 1979

REQUIREMENTS STANDARDS STUDY
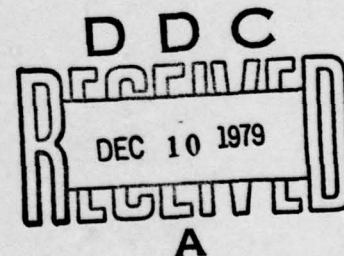## Technical Report Overview

LOGICON

Daniel G. Smith
Paul B. Merrithew

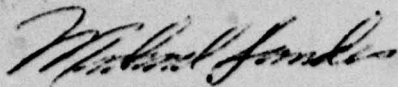APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

D D C
RECEIVED
DEC 10 1979
A

ROME AIR DEVELOPMENT CENTER
Air Force Systems Command
Griffiss Air Force Base, New York 13441

79 12 7 059

This report has been reviewed by the RADC Public Affairs Office (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.
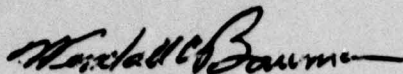
RADC-TR-79-240, Vol I (of three) has been reviewed and is approved for publication.

APPROVED:
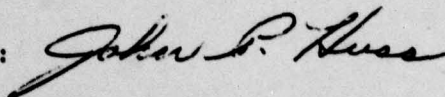
MICHAEL LANDES
Project Engineer

APPROVED:

WENDALL C. BAUMAN, COLONEL, USAF
Chief, Information Sciences Division

FOR THE COMMANDER:

JOHN P. HUSS
Acting Chief, Plans Office

If your address has changed or if you wish to be removed from the RADC mailing list, or if the addressee is no longer employed by your organization, please notify RADC (ISIE), Griffiss AFB NY 13441. This will assist us in maintaining a current mailing list.

Do not return this copy. Retain or destroy.

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER RADC-TR-79-240-VOL-1 (of three)	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) REQUIREMENTS STANDARDS STUDY. Volume I. Technical Report Overview.		5. TYPE OF REPORT & PERIOD COVERED Final Technical Report. Oct 77 — Jan 79,
		6. PERFORMING ORG. REPORT NUMBER DSJ-R79007-VOL-1
7. AUTHOR(s) Daniel G. Smith Paul B. Merrithew		8. CONTRACT OR GRANT NUMBER(s) F30602-77-C-0207
9. PERFORMING ORGANIZATION NAME AND ADDRESS Logicon, Inc. 18 Hartwell Avenue Lexington MA 02173		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 62702F 55811807
11. CONTROLLING OFFICE NAME AND ADDRESS Rome Air Development Center (ISIE) Griffiss AFB NY 13441		12. REPORT DATE October 1979
		13. NUMBER OF PAGES 46
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) Same		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

Same

18. SUPPLEMENTARY NOTES
RADC Project Engineer: Michael Landes (ISIE)

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Requirements Engineering	Software Specification
Requirements Definition and Analysis	Development Specification
System Specification	Requirements Specification Language
Functional Specification	Requirements Engineering Standards

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)
This report is one of three volumes which provide information and guidance for defining and analyzing military system/segment specification and development specification requirements (MIL-STD-490/483 (USAF)). Volume I is a technical overview describing the purpose of the study, the technical approach and a history of Air Force systems engineering management. It summarizes the results of the study: a Requirements Engineering Guidebook, an example using the Guidebook, associated automated tool capabilities, automated tool design approaches, and recommendations on implementing the Guidebook in the (Cont'd)

DD FORM 1473
1 JAN 73

UNCLASSIFIED

409 990

Item 20 (Cont'd)

Air Force systems acquisition life cycle. Volume II expands upon the material summarized in the first volume. Volume III is the Requirements Engineering Guidebook. The Requirements Engineering Guidebook describes the characteristics of good requirements, the various system requirement types, and the requirements engineering procedures. The requirements engineering procedures are described in the form of a procedural flow with accompanying guidelines and standards for performing fourteen requirements engineering activities. Each requirements engineering activity is supplemented by a description of the specific issues to be addressed during the first two phases of the Air Force acquisition life cycle - the Conceptual and Validation Phases.

PREFACE

This report is one of three volumes prepared to assist government and contractor personnel in managing and performing system requirements definition and analysis: requirements engineering. The primary results of this study has been the definition of guidelines and standards for requirements engineering (Requirements Engineering Guidebook) and the identification of automated aids to support the application of the guidelines and standards during the initial phases of the Air Force system acquisition life cycle - the Conceptual and Validation Phases.

This study reflects Logicon's experience with an automated requirements engineering tool applied in support of the acquistion of a large Air Force surveillance system. The Requirements Engineering Guidebook reflects the needs of an Air Force System Program Office acquisition environment; however, the basic requirements engineering principles and guidance are easily adapted to other acquisition environments.

This report was prepared by Logicon for the Air Force Systems Command (AFSC), Rome Air Development Center (RADC), Software Engineering Section. Administrative review and technical coordination of this report have been accomplished for RADC by Mr. Michael Landes (project officer).

Review of this report was accomplished at RADC, by Electronic Systems Division (AFSC/ESD) personnel at Hanscom, AFB, and by Logicon personnel. Special thanks to the many reviewers and for the patience and skills of Ms. Marcia Brehm and Ms. Deborah Queen for the technical typing, proofing, and revisions.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

EVALUATION

This report contains a comprehensive survey of existing standards pertinent to the requirements phase of software system development. It also contains material bearing on automated tools for improving management of this phase. This report is one of three volumes which provide information and guidance for defining and analyzing military system/segment specification and development specification requirements (MIL-STD-490/483 (USAF)). Volume I is a technical overview describing the purpose of the study, the technical approach, and a history of Air Force systems engineering management. It summarizes the results of the study: a Requirements Engineering Guidebook, an example using the Guidebook, associated tool capabilities, automated tool design approaches, and recommendations on implementing the Guidebook in the Air Force systems acquisition life cycle. Volume II expands upon the material summarized in the first volume. Volume III is the Requirements Engineering Guidebook. Volume III describes the characteristics of good requirements, the various system requirement types, and the requirements engineering procedures.

The contents of this comprehensive study comprise the main body of an Air Force Guidebook for use on software system acquisition which is of value for immediate use and for guiding research and development efforts toward generating automated tools in the requirements phase of software development.

MICHAEL LANDES
Project Engineer

1.   INTRODUCTION

Military systems acquisition has a highly regulated environment. However, many of the military system engineering management procedures are not appropriate for the complexity of the systems being procured. This volume summarizes the results of a Logicon study[1] for the Air Force in which guidelines and standards for systems requirements definition and analysis have been developed for the initial phases of an Air Force systems acquisition. The guidelines and standards are presented in the Requirements Engineering Guidebook, the third volume of this three volume report. Guidance is provided for defining and analyzing the requirements for the system as a whole (Conceptual Phase). Additional guidance is provided for expanding and refining the initial definition by the allocation of the requirements to specific components of the delivered system, with specific emphasis on computer program development specifications (Validation Phase).

This volume presents an overview of the Air Force systems acquisition environment and associated requirements engineering problems. This volume briefly discusses the methodology applied during the study and the technical results. The technical results concentrate on the Requirements Engineering Guidebook, including a description of the characteristics of quality requirements[2], a discussion of various requirements types, and the procedures for Conceptual Phase and Validation Phase requirements engineering. The Requirements Engineering Guidebook is described and

---

[1] This work is supported by F30602-77-C-0207.

[2] The term 'quality requirements' is used throughout this study to denote system requirements which are complete, consistent, testable, and traceable. This characteristic is the result of the requirements being discretely identified and well-organized as discussed in the sections to follow.

1

supported by sample output from a comprehensive automated requirements engineering tool which is currently being employed by Logicon on an Air Force project. This volume discusses the development of a list of automated-tool capabilities which support the Requirements Engineering Guidebook, concluding with a list of recommendations for further research activities.

2.　　　　AIR FORCE SYSTEMS ENGINEERING MANAGEMENT

The complexity of military systems development has continued to outpace the management and technical resources supporting the acquisition process. During the 1960s and into the early 1970s, systems development in the Air Force Systems Command (AFSC) was regulated by a series of manuals, the AFSCM 375 series. Two of these manuals concentrated on system engineering procedures (AFSCM 375-5 [1]) and documentation (AFSCM 375-1 [2]). This highly regulated approach was necessitated by the increasing delivery of obsolete systems, which resulted from the less regulated systems development approaches of previous decades (1940s-1950s). Added to this obsolescence problem was the rising complexity of the systems being developed to meet the national defense needs of the post World War II decades, including the increased application of systems with embedded software.

The AFSCM 375 series provided for flexibility in its application but was not completely understood. As a result of the difficulties encountered in applying AFSCM 375, the Air Force began to rescind parts of the series during the late 1960s. The Air Force documentation requirements of AFSCM 375-1 evolved almost unchanged into the present standards for specification practices, MIL-STD-490 [3] and MIL-STD-483 (USAF) [4]. The system engineering procedures, AFSCM 375-5 evolved into the present regulation for Air Force program office engineering (AFR 800-3 [5]). Contractor system engineering requirements are described in the present Air Force standard for engineering management, MIL-STD-499A (USAF) [6].

2

As previously reported [7], it is becoming evident that present military management practices and technical resources are not adequate for the increasingly complex military systems being developed. A principle area of deficiency has resulted from the inadequacies of the system engineering guidelines in MIL-STD-499A and AFR 800-3. In the wake of rescinding the system engineering requirements of AFSCM 375-5, significant practices for systems requirements engineering were not translated or updated into present practices. As a result many essential requirements engineering practices are non-existent or have been de-emphasized.

This trend toward less regulated Air Force systems engineering management has been encouraged by defense contractors in a desire to allow for more competitive and innovative approaches to systems development. Numerous contractors have responded by developing systems engineering procedures. However, other defense contractors and military agencies have not developed systems engineering or management practices which satisfy the real technical and management needs of Air Force programs.

In recent years systems requirements engineering has received renewed attention within academic and military research and development (R&D) environments and is now coming to the forefront of research and applications for improved military systems development. AFSC's Electronic Systems Division (ESD) has acquired a computer-aided requirements engineering tool, CADSAT, and has encouraged the application of this computer aid in Air Force requirements engineering activities.[1] Logicon

[1] The Computer-Aided Design and Specification Analysis Tool (CADSAT) is an Air-Force-owned requirements analysis tool developed by the University of Michigan under ESD/TOI contract F19628-76-C-0197. [8] [9] The extended version is a modification developed by Logicon for applications to military systems under ESD/OCU contract F19628-76-C-0218. CADSAT's User Requirements Language/User Requirements Analyzer (URL/URA) is basically equivalent to the Problem Statement Language and Problem Statement Analyzer (PSL/PSA) developed at the University of Michigan. [10]

3

systems analysts have employed CADSAT for several years in defining and analyzing the system requirements for a large Air Force surveillance system. As a result of this application, Logicon has made extensions to CADSAT to satisfy requirements engineering management and technical needs. These extensions have made CADSAT more suitable to specific military systems acquisition activities. As with other research directed in recent years toward defining standards for modern programming practices in Air Force systems development, requirements engineering is being identified as a target for improved standardization. Within this renewed interest and based upon the surveillance system experience using CADSAT, Logicon began developing a requirements engineering standard in 1977 under contract to the Rome Air Development Center (RADC), Griffiss AFB, New York. This study, titled the Requirements Standard Study (RSS), is based upon Logicon's requirements engineering experience and the use of CADSAT. The primary results of the RSS are a Requirements Engineering Guidebook (Volume III) and the identification of automated aids to support the application of the Guidebook. The Guidebook has been developed in response to the requirements engineering inadequacies of the current Air Force system engineering procedures. However, the principles of the Guidebook are applicable to other systems acquisition environments.

3.   STUDY APPROACH

The RSS was organized into the series of tasks shown in Figure 1. An extensive literature search identified existing military regulations, standards, and specifications and a variety of guidebooks and handbooks. Past and present DoD research projects were reviewed through the interactive query facility of the Defense Documentation Center. In addition, professional journals and conference proceedings were researched. As pertinent documentation was identified, bibliographic information on each was entered into a computer maintained file. The complete bibliography is included in Volume II of this report.

4
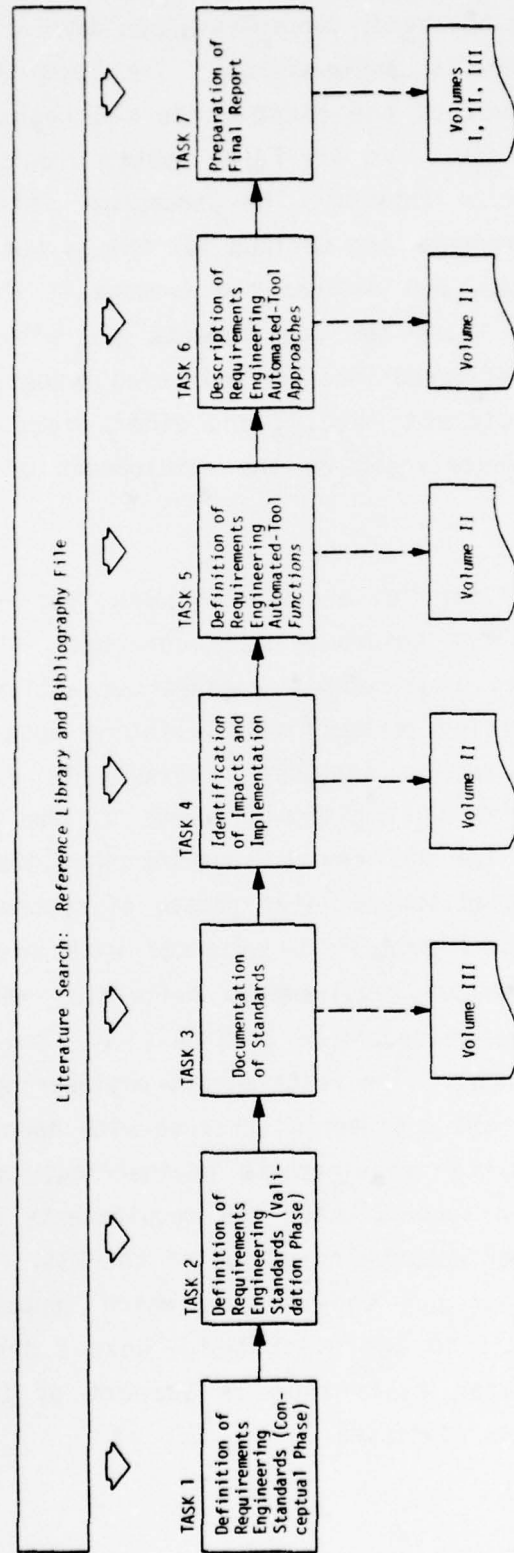
Figure 1. Requirements Standards Study Technical Approach

The principle tasks of the study concentrated on developing guidelines and standards for requirements engineering. The Requirements Engineering Guidebook was a product of the first three RSS tasks. The Guidebook concentrates upon the first two Air Force system acquisition phases, the Conceptual and Validation Phases. The Conceptual Phase is the initial period when the requirements are defined for the system as a whole. The Validation Phase expands and refines the Conceptual Phase requirements. During the Validation Phase the requirements are allocated to specific end-items which are configured into the delivered system, such as hardware (radars, computer equipment, etc.), and other items such as computer programs. The RSS concentrated on the development of computer program requirements.

As the preparation of the Guidebook continued, the role of automated assistance to support the Guidebook was addressed. In developing the Guidebook the necessary requirements engineering activities were viewed from a systems engineering perspective. Existing automated tools which have evolved from academic and R&D environments lack many of the fundamentals of requirements engineering needs of the Air Force systems engineering process. The RSS review suggests that these initial tools were designed with a bias toward later phases of systems development and are burdened by the same problem experienced with most other software systems: existing automated requirements definition and analysis tools are attempting to solve an undefined problem (i.e., the requirements for the tools are ill-defined). The requirements engineering process must be defined within the systems engineering process with specific attention to the early phases of system requirements engineering, then the study of automated assistance in accomplishing the requirements engineering tasks may proceed. The latter was the objective of the RSS. As a result, the requirements engineering tool capabilities which support the Guidebook were separately identified and described. Next a description of two approaches for automated assistance in support of the Requirements Engineering Guidebook was addressed.

6

This final technical report contains the following: the Requirements Engineering Guidebook; a description of automated tool capabilities which can support the Guidebook; examples of the use of an existing automated tool (Logicon-Extended CADSAT) using the Guidebook; a description of two approaches for applying CADSAT in support of the Guidebook; recommendations on implementing the Guidebook within the existing regulations, applicable standards and specifications, and other results and recommendations.

4.   REQUIREMENTS ENGINEERING GUIDEBOOK

Requirements Engineering

During this study, requirements engineering was determined to be a distinct engineering discipline which needs to be addressed separately from other aspects of systems engineering. During the 1960s requirements engineering was integral to the procedural aspects of the system engineering process established under AFSCM 375-5. Neither the current military standard for engineering management (MIL-STD-499A) nor the guidance for program office engineering (AFR 800-3) define requirements engineering. Requirements engineering is vaguely defined to be part of the system engineering process: the functional analysis-synthesis tasks. This type and form of guidance is inappropriate for the requirements engineering activities which must be accomplished during the early phases of the acquisition process. A requirements engineering definition must be stated and the procedural issues addressed. The following definition has been prepared during the course of the RSS:

Requirements Engineering is an iterative process of defining the system [1] requirements and analyzing the integrity of the requirements. This process involves all areas of system development preceding the actual design of the system. The products of the requirements engineering process can be evaluated for completeness, consistency, testability, and traceability. The essential goal of Requirements Engineering is to thoroughly evaluate the needs which the system must satisfy.

[1] A system in the context of this presentation is an aggregate of equipment, personnel resources, capabilities and techniques which collectively perform an operational role. The composite system includes all related facilities, items, materials, services, and personnel required for the system's self-sufficient operational deployment.

7

This definition distinguishes requirements engineering from other engineering management tasks such as program planning, costing, trade-off studies and a host of other issues surrounding the early phase of systems development. The definition distinguishes requirements engineering as being concentrated upon the actual definition of the system requirements.

The lack of specific approaches and techniques for military requirements engineering allows even the best intentioned analyst to digress rapidly from the "need" category to the "how-to" or solution oriented requirements definitions. This is a natural tendency especially for any design-oriented engineer, such as a software engineer. During the course of requirements engineering, the analyst must also be aware that non-design-oriented system documentation, such as functional (Type A, MIL-STD-490/483 (USAF) System/Segment Specifications) and development (Type B, MIL-STD-490/483 (USAF)) specifications, is the medium for communicating the system requirements to the design engineers. The requirements engineering goal is to identify "discrete" requirements of the system and to organize these requirements in effective ways for further analysis. The results of this process is a set of "quality requirements."

Quality Requirements Characteristics

A set of quality requirements consists of discrete requirements, well organized to permit further analysis Early requirements documents usually have one prevailing characteristic: the requirements are spread over various source documents and/or presented in various parts of the documents, and the requirements overlap each other. This is partly because of the fragmented nature of the early planning and study efforts which are formulative and investigatory. The specification documents in many instances are products written to meet acquisition needs and schedules rather than repositories of quality system requirements.

Figure 2 illustrates the first characteristic of quality requirements: discreteness. The key to identifying discrete requirements is to break the

8

DISCRETE & WELL-ORGANIZED REQUIREMENTS

DISCRETE REQUIREMENTS

REQUIREMENTS IN SOURCE DOCUMENTATION
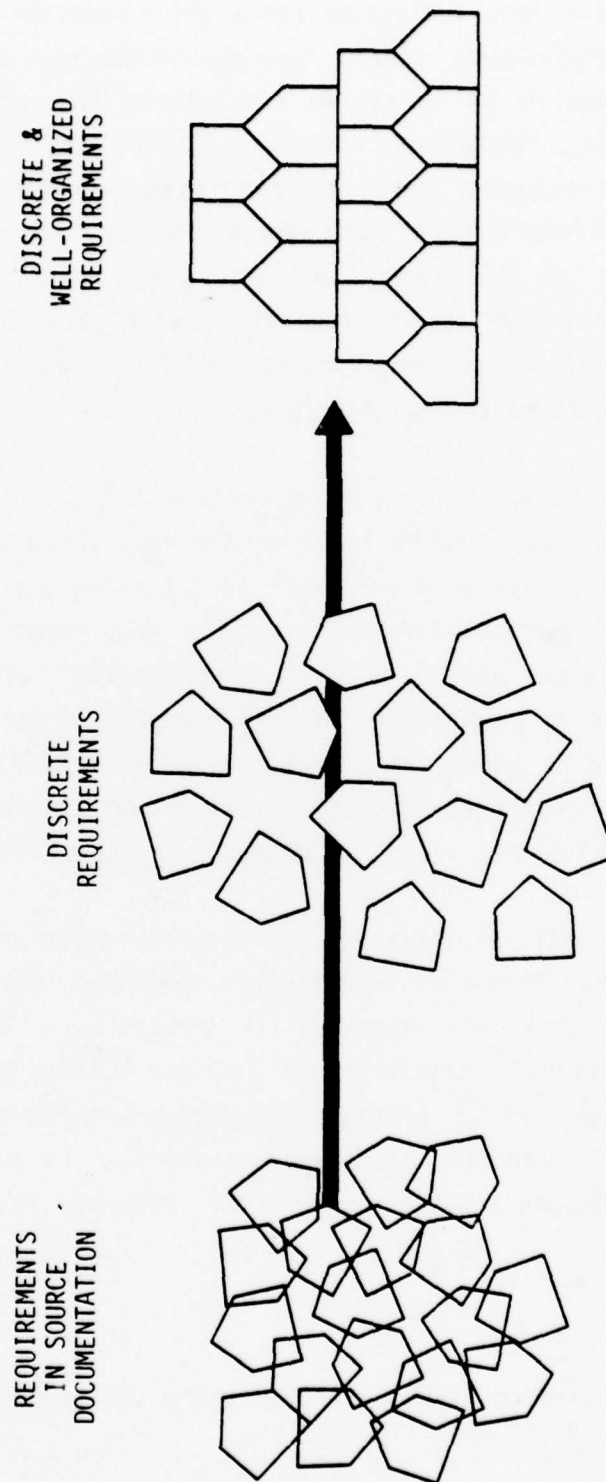
Figure 2. Development of Discrete and Well-Organized Requirements

9

source documentation into individual parts which represent non-overlapping requirements. Requirements should then be categorized as functions the system must accomplish or as system constraints (performance, physical, operability, test, design).[1] At this point missing or incomplete requirements can be more readily identified. This itemization and categorization of requirements introduces clarity, where as the source documentation may be overstated, ambiguous, redundant, incomplete, and inconsistent. This process of itemization also provides the basis for verifying the quality of the requirements and for accessing the ability to test the requirements in the target system.

The second characteristic of a good statement of requirements is the arrangement of the requirements in effective ways for additional analysis and for communicating these requirements to the using agency and to design engineers. The identification of discrete requirements provides some awareness of omissions and gaps in the requirements. This awareness is further heightened by organizing the requirements in ways which identify all the relationships among the discrete requirements (Figure 2). These relationships are of three types: logical organizational relationships, system flow relationships, and requirements traceability relationships.

Logical organizational relationships are shown by structuring the discrete functions and the information requirements (external and internal input/ output) of the system into hierarchical structures. The concept of a functional hierarchical structure was introduced into military systems development through initial systems engineering practices dating back to the 1940s. This concept has been maintained in military systems development and documentation throughout the 1960s and is an integral part

---

[1] The system requirement types (functions and constraints) are discussed in more detail in the next few pages.

of the current military standards for system documentation, i.e.,
MIL-STD-490 [3], MIL-STD-483 (USAF) [4], DoD 7935.1-S [11]. Current
techniques for system development, such as the Hierarchical Input-Output-
Process (HIPO [12]) visual table of contents and automated requirements
analysis tools (PSL/PSA, CADSAT) retain the principles of functional
hierarchical structures. This form of organization provides a view of the
system as an aggregate of functions broken into a logical arrangement of
subordinate discrete activities which must be performed. A sample portion
from the Logicon-Extended CADSAT Structure Report (Figure 3) demonstrates
the functional break out of a space system[1]. This section of the report
shows the hierarchical breakdown of the space-system-x into discrete
functions. Each breakdown of the functions is denoted by the indented
format and the hierarchy level number. For example, boost breaks down into
four level 4 subfunctions. Over the course of requirements engineering,
many missing or incomplete functions can be directly identified from the
functional hierarchical structure.

The discrete system inputs, outputs (external I/O) and the internal
information requirements necessary for the system's operation can be
logically structured in the same manner as the functional hierarchy. The
emphasis again is the arrangement of the information requirements into
structures by breaking the information into logical subordinate parts or
simply as groupings. A well-organized structure is effective in
communicating the information requirements and for identifying incomplete
or missing information requirements.

System flow relationships can be shown by organizing the discrete
requirements in terms of control flow and information flow. As the
functions of the system are defined, the control relationships between
them are identified. These control relationships describe the logical

[1] Figures 3, 4, and 5 are CADSAT-like reports based upon the
     space-system-x example contained in AFSCM 375-5, attachment 2, pp
     128-130 [1]

11

LINE NO.	HIERARCHY LEVEL NO.	REQUIREMENT NAME	REF. DOCUMENT PARAGRAPH NO.
1	1 space-system-x		3.1.1
2	2 status-monitoring		3.1.1
3		triggers launch	3.1.1
4	2 launch		3.1.1
5		triggers flight-mission	3.1.1
6	2 flight-mission		3.7.3
7		triggers boost	3.7.3.1
8		triggers steer	3.7.3.1
9	3 boost		3.7.3.1
10		triggers release-payload	3.7.3.1
11		triggers stage-1-and-stage-2-thrust	3.7.3.1
12	4 stage-1-and-stage-2-thrust		3.7.3.1.4
13		triggers jettison-stage-1	3.7.3.1.4
14	4 jettison-stage-1		3.7.3.1.4
15		triggers stage-2-thrust	3.7.3.1.4
16	4 stage-2-thrust		3.7.3.1.4
17		triggers stage-2-shut-down	3.7.3.1.4
18	4 stage-2-shut-down		3.7.3.1.4
19	3 steer		3.7.3.1
20		triggers release-payload	3.7.3.1
21	3 release-payload		3.7.3.1
22		triggers decelerate-launch-vehicle	3.7.3.1
23	3 decelerate-launch-vehicle		3.7.3.1
24		triggers payload-coast	3.7.3.1

Figure 3. Space System X: Functional Hierarchical Structure

order in which the system activities should be accomplished to satisfy the system mission and operational requirements. Figure 4 is a control-flow report for a portion of the space-system-x. In this report (CADSAT Process Chain) the flow of control is from left to right. Any number of CADSAT process chain reports can be generated to provide the analyst with a comprehensive understanding of the system control flow. Control-flow analysis provides a means of viewing the system from an activity-oriented perspective and is often referred to as functional-flow analysis. On the other hand, information-flow analysis builds upon the information hierarchy structure by providing a means of viewing the system as an information processing system. During this analysis the flow relationships between external system inputs and resulting outputs are identified. Quite often the most effective means of performing information-flow analysis is to trace an output back to system inputs: external data, messages, or stimuli. As a result of this analysis the relationships between the associated functions and the internal information necessary to support the derivation of the output are identified. Control-flow and information-flow analysis will identify necessary changes and additions to previously defined functions and constraints as well as to the hiearchy structures and other relationships. Missing or incomplete requirements can be determined and the deficiencies corrected.

Requirements traceability analysis provides the analyst with a means of verifying the requirements by linking each requirement to all forms of source documentation. The Requirements Traceability Report (Figure 5) shows the traceability between specifications contained in separate requirements data bases. Figure 5 traces the requirements one specification of the space-system-x to the allocated requirements contained in the next level of specification, This form of analysis aids in validating the requirements. Relationships can also be defined to other pertinent studies, analyses, and plans which are being accomplished concurrently with the requirements engineering activities. The links to associated system plans, analyses, and studies accomplished prior to, during, and subsequent to the start of formal requirements engineering are

FLOW OF CONTROL IS FROM LEFT TO RIGHT

```
                                             +------------+
                                             :provide-    :........ +-----------+
                                             :separation- :         :energy-    :
                                             :command     :         :release-   :
                                             +------------+         :stage-1    :
                                            *              *        +-----------+
                                           *                *
                              +------------+                 +------------+
                              :sense-sep-  :                 :no-go-to-gd :        +-----------+
                           ** :signal-fm-  :                 :-spacecraft :....... :dsp-fail-in:
                              :auto-sys    :                 :-cntl       :        :spacecraft-:
                              +------------+                 +------------+        :cntl       :
                             *                                                     +-----------+
                            *                                        *
                           *                                          +-----------+
              +------------+                                          :dsp-fail-at:
              :shut-       :                  +----------+            :ground-    :
           ** :down-       :                  :          :            :cntl       :
              :stage-1     :                  :  abort   :            +-----------+
              +------------+                  :          :
                           *                  +----------+
                            *                ** 
                             *                                        +-----------+
                              *                +------------+         :loops to   :
                               +------------+  :sense-sep-  :........ :previous   :
                            ** :null-stage- :  :signal-fm-  :         :entry      :
                               :2-flight-   :  :auto-sys    :         +-----------+
                               :cntl        :  +------------+
                               +------------+
                                           *
                                          *
                              +------------+
                              :sense-sep-  :
                              :signal-fm-  :
                              :auto-sys    :
                              +------------+
```
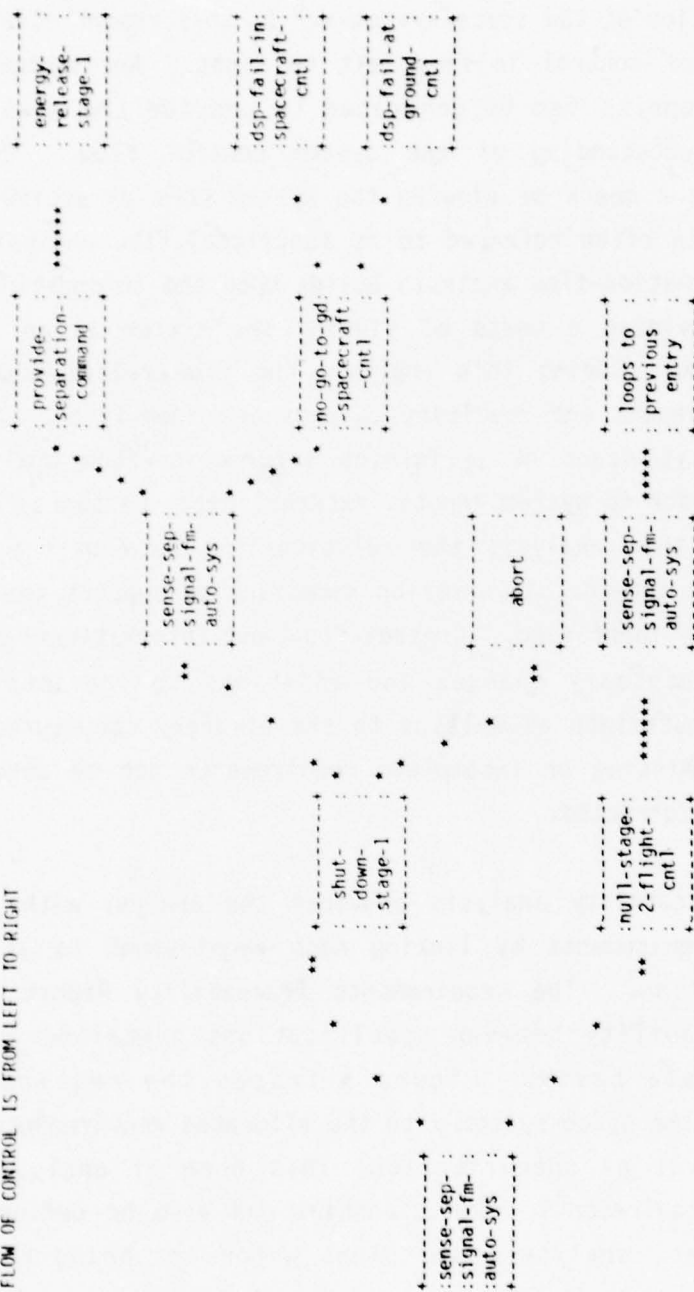
Figure 4. Space System X: Control-Flow Diagram

PRIMARY DOCUMENT PARAGRAPH NUMBER	PRIMARY REQUIREMENT NAME	SECONDARY DOCUMENT PARAGRAPH NUMBER	SECONDARY REQUIREMENT NAME
3.7.3.1.4	jettison-stage-1	3.2.3.1	provide-primary-automatic-staging-signal
3.7.3.1.4	jettison-stage-1	3.2.3.4	provide-auto-2nd-staging-signal-at-final-commit-time
3.7.3.1.4	jettison-stage-1	3.2.3.5	provide-auto-2nd-staging-signal-at-programmed-accel
3.7.3.1.4	jettison-stage-1	3.2.3.2	sense-stage-separation-signal-from-auto-systems
3.7.3.1.4	jettison-stage-1	3.2.3.6	no/go-annunciator-to-ground-and-spacecraft-control
3.7.3.1.4	jettison-stage-1	3.2.3.3	display-fail-at-ground-control
3.7.3.1.4	jettison-stage-1	3.2.3.9	ground-remote-manual-signal
3.7.3.1.4	jettison-stage-1	3.2.3.1	display-fail-in-spacecraft-control
3.7.3.1.4	jettison-stage-1	3.2.3.10	spacecraft-remote-manual-signal
3.7.3.1.4	jettison-stage-1	3.2.3.5	read-manual-signal-into-logic-stream
3.7.3.1.4	jettison-stage-1	3.2.3.9	null-stage-2-flight-control
3.7.3.1.4	jettison-stage-1	3.2.3.8	shut-down-stage-1
3.7.3.1.4	jettison-stage-1	3.2.3.2	abort
3.7.3.1.4	jettison-stage-1	3.2.3.6	provide-separation-command
3.7.3.1.4	jettison-stage-1	3.2.3.4	energy-release-and-unfasten-stage-1
3.7.3.1.4	jettison-stage-1	3.2.3.10	jettison-stage-1-from-launch-vehicle
3.7.3.1.4	jettison-stage-1	3.2.3.3	monitor-stage-1-separation
3.7.3.1.4	jettison-stage-1	3.2.3.7	provide-separation-complete-signal
3.7.3.1.4	jettison-stage-1	3.2.3.4	provide-on-command-to-stage-2-flight-control
3.7.3.1.4	jettison-stage-1	3.2.3.2	stage-2-thrust

Figure 5. Space System X: Requirements Traceability Report

15

crucial to the overall systems engineering concept. Throughout the requirements engineering activities the analyst must be able to evaluate the impact of changes to the requirements. Once the area of impact is identified in the requirements engineering products (Functional and I/O hierarchies, control and information-flows etc.) the traceability relationships provide the capability to readily identify associated impacts to the system and to trace the impacts to all other associated documentation. The impact can be readily analyzed and the appropriate actions taken.

Discrete and well-organized requirements support the primary goal of defining the operational mission needs of the using activity while giving the analyst visibility and control over the system definition process. Discrete and well-organized requirements are prerequisites for the creation of system functional or development specifications.

System Requirement Types

Understanding the various system requirements types and their use contributes significantly to the identification of discrete requirements and, therefore, quality requirements definitions. Table 1 shows that there are two sets into which system requirements may be organized.

The functional requirements set is the backbone of the system requirements engineering process. It is within this set of requirements that the pure design-free or solution independent needs are declared. Simply stated, and the functional requirements represent the total discrete system activities required to achieve a specific objective. A functional requirement identifies what must be accomplished without identifying any aspect concerning the means such as hardware, computer programs, personnel, facilities, or procedural data. The functional requirements represent a problem statement devoid of any overtone or specifics regarding real or conceptual solutions which satisfy any or part of the needed functions.

16

Table 1. System Requirement Types

SYSTEM REQUIREMENTS	FUNCTIONAL REQUIREMENTS (functions)		The set of discrete functions which identify the pure design free or solution independent needs of the system as a whole. The functional requirements identify what must be accomplished while avoiding solution statements or overtones.
	CONSTRAINT REQUIREMENTS (Constraints)	PERFORMANCE	How well the system functions must be accomplished, such as timeliness and accuracy. Also called performance characteristics, MIL-STD-490.
		PHYSICAL	Influences the design solution in a physical manner: power, size, weight, environment, human factors, existing system interfaces, GFP, etc. Also called Physical Characteristics, MIL-STD-490.
		OPERABILITY	Reliability, maintainability, availability, dependability.
		TEST	Identify the functional, performance, physical, operability, and design requirements which will be evaluated during system integration and test.
		DESIGN	The minimum or essential design and construction requirements which are a constraint on the functional requirements of the system during the design and construction of the system end-items (CIs/ CPCIs). Also called Design and Construction, MIL-STD-490.

17

Some examples of discrete top-level functions for an electronic system might be surveillance, tracking, identification, interceptor control, and communications.

The second set of requirements is the constraint set which consists of five requirements types: performance, physical, operability , test, and design (as described in Table 1). The constraint set modifies the functional requirements set. Without the constraint set, a solution for the system functional requirements could not be achieved. However,excessive or unrealistic constraints can eliminate all solutions or increase the technical risks and cost of the solution. Therefore, the identification and management of the constraint requirement set must be achieved with care. Whenever specific constraints are identified, there must be sufficient justification, such as an engineering analysis, which clearly shows that the constraint is a reasonable, necessary, and practicable, and represents an actual requirement and not just a desirable feature.

Requirements Engineering Procedure

Requirements engineering is an _iterative_ process of defining the system requirements and analyzing the integrity of the requirements for completeness, consistency, testability, and traceability. As the process continues, the system requirements are defined and analyzed in a progressively expanding manner. The definition and analysis activities will move from one area of concentration to another as the results of previous activities reveal areas needing additional work. No singular approach can be rigidly defined and applied which can take into account the many possibilities which must be considered. However, guidelines for requirements engineering and associated tasks can be defined and then tailored for specific requirements engineering applications. The following is a synopsis of the requirements engineering procedures contained in Volume III, Requirements Engineering Guidebook. The general framework for requirements engineering is illustrated in Figure 6. Each block represents a unique requirements engineering activity which must be accomplished in
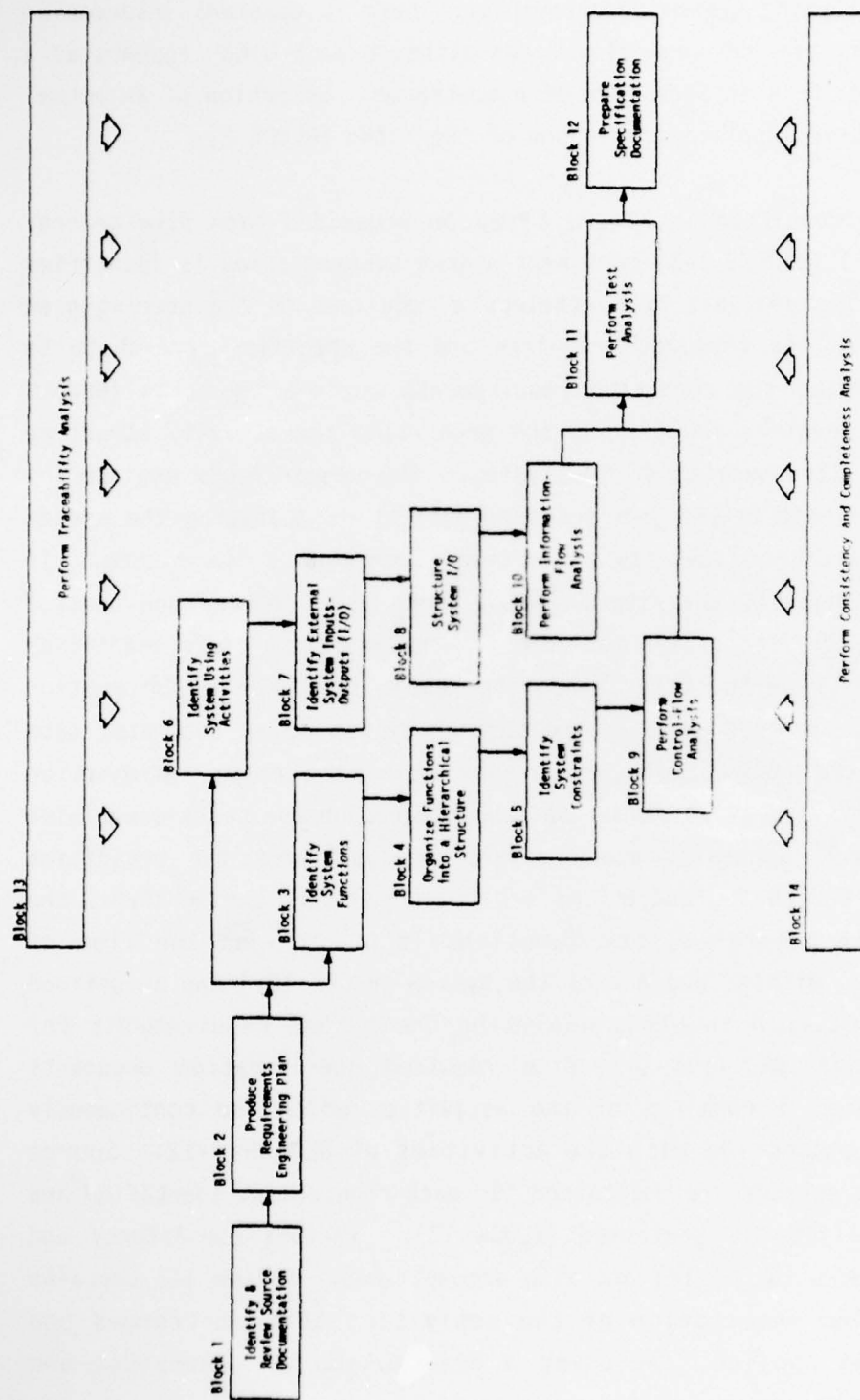
18

Figure 6. Requirements Engineering Procedures

19

defining and analyzing system requirements. There is constant interaction between the activities of each block, and although each block appears as a single activity, it is in fact part of a continuum. Selection of an actual approach for a given application is one of the tasks (BLOCK 2).

The activities identified in Figure 6 may be organized into five general steps. In step 1 (BLOCKS 1-2) pertinent source documentation is identified and reviewed. The analysis team develops a requirements engineering plan which identifies the resources required and the specific approach to be taken in performing the remaining requirements engineering tasks (BLOCKS 3-14). Step 2 involves identifying and organizing the activity structure and information structure(s) of the system. The requirements engineering tasks associated with BLOCKS 3-5 are concentrated on analyzing the system source documentation to identify activities performed by the system. If the system is primarily activity oriented, such as a command and control system, the analysis activities may be concentrated on the tasks identified in BLOCKS 3-5. If on the other hand, the system is primarily information oriented, as in the case of a communications system or an automated data processing system (ADP) application such as a management information system, the analysis activities may be concentrated on the tasks associated with BLOCKS 6-8. Generally the analysis team performs the activities associated with BLOCKS 3-5 and BLOCKS 6-8 concurrently. During step 3 the flow of control between system functions (BLOCK 9) and the flow of information into, within, and out of the system (BLOCK 10) can be defined and analyzed. Step 4 involves analyzing the system requirements for testability (BLOCK 11) and preparing required specification documents (BLOCK 12). Step 5 consists of two activities which are continuously performed in conjunction with the activities of BLOCKS 3-12. Source documentation references are maintained for each requirement identified and traceability analysis is performed (BLOCK 13). Various consistency and completeness checks (BLOCK 14) are also accomplished. Volume III contains a more detailed description of the activities to be performed and standards to be applied, including a description of Conceptual and Validation Phase issues.

5.　　　　　REQUIREMENTS ENGINEERING TOOL CAPABILITIES

The following paragraphs describe the role of automation in requirements engineering and summarizes a more detailed discussion of automated tool capabilities presented in Volume II.

Intrinsic Capabilities of Automated Tools

Automated tools like CADSAT assist requirements engineering in four ways:

- Provide a medium for formal requirements definition

- Perform rudimentary analysis

- Produce documentation

- Permit a flexible, iterative approach to requirements definition

Automated tools,like CADSAT, consist of two parts:　a language and an analyzer.　The language provides the means for describing the requirements for functional and development specifications.[1]　The report language and analyzer can be used to assist the analyst in completing the tasks described in the Requirements Engineering Guidebook (Volume III).

---

[1]　In Air Force system acquisitions the functional specification is the system/segment specification (Type A, MIL-STD-483 (USAF), Appendix III) and the development specifications are Type B specifications.　The Computer Program Configuration Item Specification (Type B5, MIL-STD-483 (USAF), Appendix VI) is the primary development specification addressed in this study.
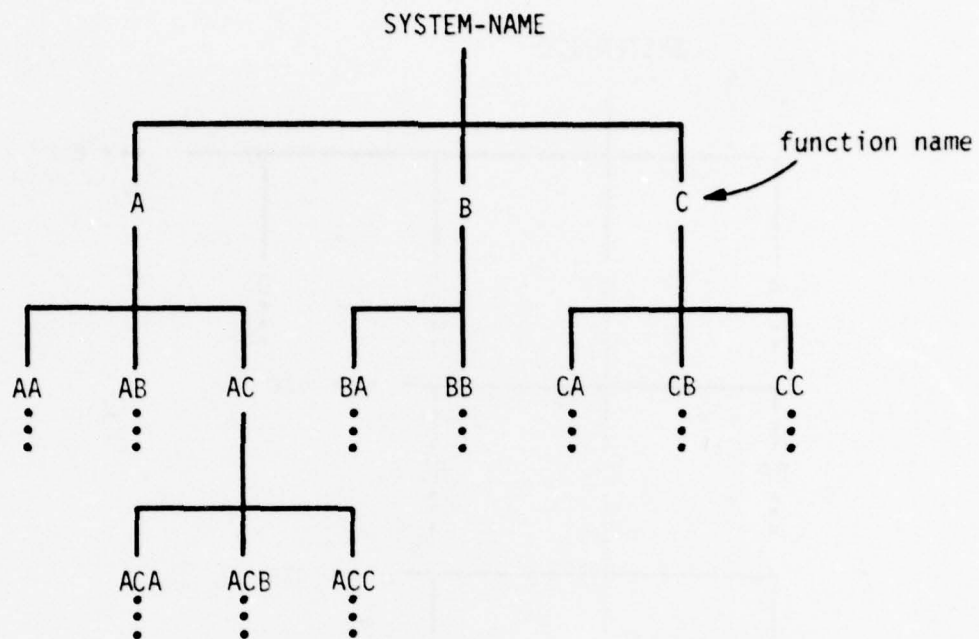
Language Objects and Relationships

The language objects and relationships described in this paragraph incorporate all the system requirements and provide the means to analyze the requirements through automated means. The "nouns" of a requirements definition language are called objects. For example, there are objects for describing system functions and other objects for describing system external and internal inputs and outputs. Each object is named. For example, the requirement, "sense stage preparation signal from automatic systems," is a functional requirement and might by entered in the requirements data base as a function object called:

sense-stage-separation-signal-from-auto-systems.

Depending on the application of the automated tool, not every aspect of the requirements has to be formalized. The essential objective is to make the requirements discrete and to organize the requirements as a basis for further analysis.

The language should allow various relationships between objects to be described. These are the "verbs" of the language. Several relationships describe simple requirement-to-requirement and requirement-to-document associations. For example, certain relationships establish the hierarchical structure of functional requirements (Figures 3 and 7) while others define the hierarchical structure of system external and internal inputs and outputs (Figure 8).

Some relationships describe the flow of control among functions (Figure 9). Some of these control flow relationships are also illustrated in Figure 3. Other relationships describe the flow of information into, within, and out of the system (Figure 10). Each requirement object (function, constraint, I/O, etc.) and relationship (functional and information (I/O) hierarchy, control and information flow, etc.) can be supplemented by a textual description.

22

```
                        SYSTEM-NAME
                             |
        ┌────────────────────┼────────────────────┐          function name
        |                    |                    |
        A                    B                    C ◄─────
        |                    |                    |
   ┌────┼────┐          ┌────┼────┐          ┌────┼────┐
   |    |    |          |    |    |          |    |    |
  AA   AB   AC         BA   BB              CA   CB   CC
  ⋮    ⋮    |          ⋮    ⋮               ⋮    ⋮    ⋮
       ┌────┼────┐
       |    |    |
      ACA  ACB  ACC
       ⋮    ⋮    ⋮
```

<u>Graphic Representation</u>

```
SYSTEM-NAME
    A

            AA ...
            AB ...
            AC ...

                    ACA ...
                    ACB ...
                    ACC ...

    B
            BA ...
            BB ...
    C
            CA ...
            CB ...
            CC ...
```
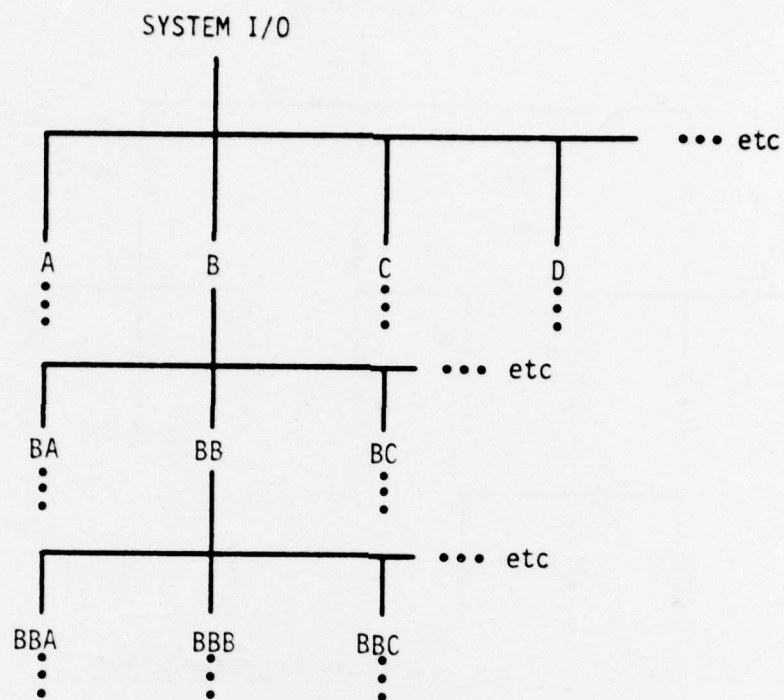
<u>Indented Representation</u>

Figure 7. Functional Hierarchical Structure

SYSTEM I/O

```
                    |
    _____|_____  ••• etc
    |               |               |           |
    A               B               C           D
    •               •               •           •
    •               •               •           •
                    |
            _____|_____  ••• etc
            |       |       |
            BA      BB      BC
            •       •       •
            •       •       •
                    |
            _____|_____  ••• etc
            |       |       |
            BBA     BBB     BBC
            •       •       •
            •       •       •
```

Graphic Representation

```
SYSTEM I/O
    INPUT-A ...
    OUTPUT-B

        BA ...
        BB
            BBA ...
            BBB ...
            BBC ...
           (etc)
        BC ...
       (etc)
    INPUT-C ...
    OUTPUT-D ...
    (etc)
```
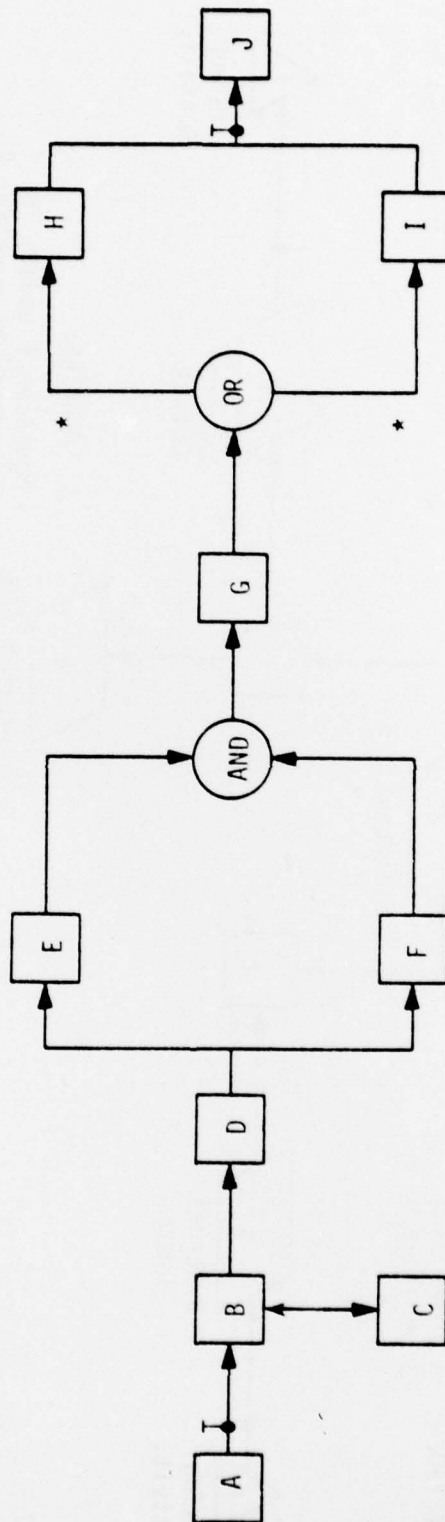
Indented Representation

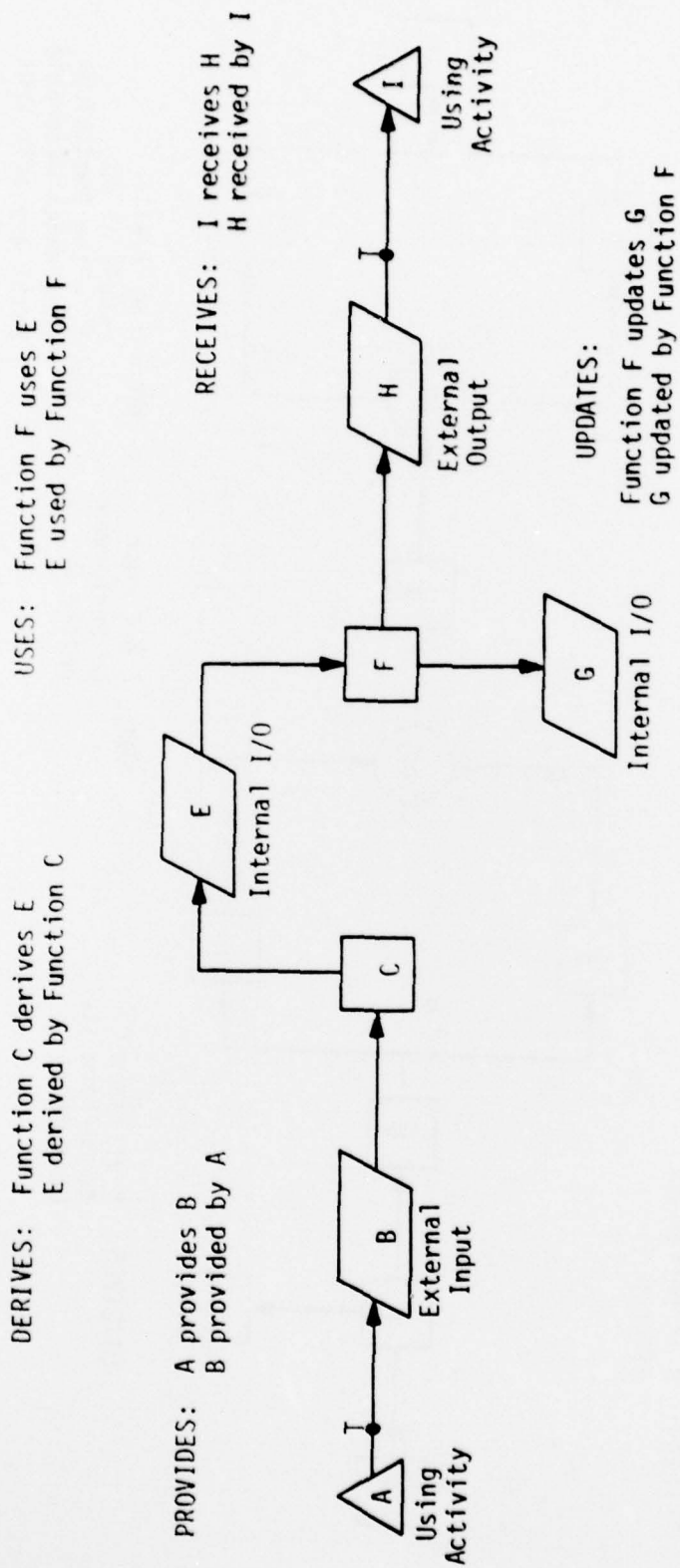Figure 8. I/O Hierarchical Structure

SERIES: B is performed
          after A

UTILIZES: B utilizes C
           to perform its
           activities

T = Test Point

AND: E & F must
      be performed
      before G

OR: H or I will
     result in J;
     * the conditions
       upon which alternate
       paths are selected

Figure 9. Control-Flow Diagram

25

DERIVES: Function C derives E
E derived by Function C

USES: Function F uses E
E used by Function F

PROVIDES: A provides B
B provided by A

RECEIVES: I receives H
H received by I

UPDATES:
Function F updates G
G updated by Function F

T = Test Point

Figure 10. Information-Flow Diagram

26

The Analyzer

The analyzer is the second part of an automated tool like CADSAT. It generates a series of reports. The reports, essential for the application of the Requriements Engineering Guidebook, can be grouped into six general report categories: requirements data base management, functional analysis, I/O analysis, traceability analysis, test analysis and documentation.

Requirements Data Base Management

Change Requirements Data Base Reports - The analyzer handles requirements definition entries into the requirements data base and changes definitions already in the data base. The report is in the form of a listing of changes made to the requirements data base.

Object Information Report - The object information report is used to check the contents of the requirements data base. Provided with a list of object names, the report supplies a listing of selected information about each object.

Source Document Summary Report - The Source Document Summary report is used to compare the requirements data base contents against the source documentation. The report presents a sequential list of all source document references.

Identify Specified Objects Report - The purpose of this report is to retrieve requested object and relationship information from the requirements data base. It relieves the analyst of simple but time consuming tasks. For example, this report aids in finding sources which have not been completely analyzed and referenced or functions which have no control or information flow relationships.

Requirements Data Base Status Reports - The Requirements Data Base Status Reports provide summary information on the contents of the requirements

data base. Requirements data base objects are listed along with various statistics showing the quantities, percentages (as appropriate), and quality of each object in the requirements data base.

Functional Analysis

Functional Hierarchical Structure Report - The primary purpose of this report is to provide requirements visibility. The report uses the functional hierarchical structure information contained in the requirements data base to present the breakdown of system functions from the general to the specific. The secondary purpose of the Functional Hierarchical Structure report is to present requirements data base information in a format that is easily used by the analyst.

Control-Flow Report - The Control-Flow report helps identify the completeness and consistency of system control flow. On input of a function name, the report traces the control flow forward or backward by a specified number of functions. Missing control flow logic is highlighted by a premature termination of the flow sequence in the report.

I/O Function Interaction Report - The I/O Function Interaction report shows the information flow for selected functions or I/O. The report is useful when the analyst is concerned with a portion of the system relative to a selected group of I/O. It answers such questions as "How does the system I/O tie these functions together?" or "Where does this I/O fit into the system?"

I/O Analysis

I/O Hierarchical Structure Report - This report prints selected parts of the I/O structure. The report is used by the analyst to review and upgrade the I/O structure. The final results provide visibility into the system I/O structure.

Information-Flow Report - Information-Flow reports help to assure a complete and consistent I/O description of the system. The report can also be prompted to trace system I/O from the system external inputs toward the external outputs (or vice versa). The report helps the analyst to examine the information flow for logical errors and inconsistencies. When the report is unable to trace back to system external inputs, missing functions or flow relationships are indicated.

Traceability Analysis

Find Related Requirements Report - This report aids change impact analysis by using the requirements data base information to locate requirements which are in some way related to a requirement which may be changed.

Requirements Traceability Report - The Requirements Traceability report shows the traceability of requirements from one set of documentation to another. Various options are provided. Requirements are traced from the source documentation to the requirements data base (based on a second set of source documentation) or from the source documentation to the second requirements data base.

Test Analysis

Test Reports - The test reports are used to evaluate the quality and completeness of test plans and procedures. Reports can be prepared for each test case defined for the information and control flows. The test reports show the relationship between system flow test points and associated test cases, test plans and procedures and other pertinent source documentation.

Documentation

Requirement Document Reports - The Requirement Document Reports are

automated reports which can be used directly in system documentation. These reports should conform to the format requirements of the prescribed documentation standard, such as MIL-STD-490/483 (USAF).

6.   ADDITIONAL STUDY RESULTS

The primary results of the Requirements Standards Study are the Requirements Engineering Guidebook and the description of automated tool capabilities. Additional results include: an example illustrating application of the Guidebook in an Air Force system acquisition, a description of an implementation approach for applying the Guidebook in Air Force acquisitions, and a discussion of two approaches for employing CADSAT in the requirements engineering activities presented in the Guidebook. These additional results are summarized below.

Requirements Engineering Example

The requirements engineering example presented in Volume II, Appendix E was derived from actual requirements engineering activities associated with an Air Force surveillance system acquisition. Excerpts from the surveillance system segment specification (Type A) are included at the conclusion of the example. The example presents a description of the actual requirements engineering performed on the specification in conjunction with the use of an automated requirements tool, Logicon-Extended CADSAT. Cross references between the example description and the requirements engineering activities in the Requirements Engineering Guidebook are identifed by references to appropriate activities.

Implementation Approach

The implementation of the Requirements Engineering Guidebook must address certain issues, practices, and policies within the Air Force systems acquistition life cycle. The present lack of direction provides little visibility into the requirements engineering activities of Air Force

program office engineering staffs or their support engineers. The principle program office goal is typically to produce a draft system/segment specification which satisfies program office schedule and milestone objectives. The current specification preparation process is generally an iterative drafting-review-redrafting process which takes place over many months. As a result, discrete and well orgainzed requirements are not apparent and are seldom easy to identify in the resulting specification documents. The Requirements Engineering Guidebook, like the concept of AFSCM 375-5, places the emphasis on engineering tasks preceding the preparation of specifications. The various forms of intermediate documentation required are more suitable to the needs of identifying recording, and communicating the requirements as they evolve. Implementation of the Guidebook must, on the one hand, recognize that intermediate documentation provides the necessary visibility leading to the preparation of good system requirements documents (Type A and B5 specifications) while also recognizing that automated assistance is necessary to reduce the burden in production and maintenance of required intermediate and final system documentation.

The introduction of the Guidebook in the Air Force program office environment for use in preparing or managing the preparation of specifications will necessitate training in order to be successful. Although a requirement to apply the Guidebook could be directed towards program offices, the lack of training coupled with a certain resistance factor could result in the Guidebook being utilized only minimally, thereby lessening the benefits which the Guidebook can provide. The training must encourage a positive attitude and assist engineers in performing their work.

Application of the Guidebook will require changes in current regulations, standards, and specifications. However, these changes are minor. They are essentially refinements and clarifications. The concept of specification development (Type A, Type B, etc.) is a well established procedure for successive refinement of "needs" which leads to a "design

to" concept and ultimately to the realities of an "as built" product. The Requirements Engineering Guidebook is supportive of this system acquisition concept. However, the required formats for specification preparation within these current standards could be improved, especially with consideration to automated documentation and specification generation from computer maintained requirements data base.

Various Air Force quality assurance requirements and guidelines would also require changes. The Requirements Engineering Guidebook intermediate documentation requirements provide quality assurance personnel with the ability to evaluate the progress of systems requirements definition and analysis, and to ensure that the requirements are clearly stated and unambiguous. Changes to current quality assurance tasking should require a more active role in the requirements engineering activities and review of intermediate documentation.

Changes are also necessary to the engineering management concepts required in MIL-STD-499A (USAF) and AFR 800-3. MIL-STD-499A describes the fundamental concepts and criteria against which contractors can propose their individual internal procedures as a means of satisfying Air Force engineering requirements. It does not specifically address requirements engineering. The definition of requirements engineering should be incorporated into MIL-STD-499A. Other changes should direct the contractor to address requirements engineering in the System Engineering Management Plan (SEMP). Finally, the system program office acquisition management guidance (AFR 800-3, Engineering for Defense Systems) should be modified in conjunction with MIL-STD-499A. Again the definition of requirements engineering should be incorporated into AFR 800-3, and specific direction for the program office to perform requirements engineering should also be included as a separate acquisition management task.

Automated Tool Design Approaches

The requirements engineering tool capabilities described in Section 5 represents the minimum capabilities necessary to support the Requirements

32

Engineering Guidebook. Each tool capability has been evaluated against the basic CADSAT capabilities as well as the Logicon-Extended CADSAT. Basic CADSAT satisfies most of the standard requirements engineering needs, especially the language capabilities. General deficiencies are in the report generation area. These deficiencies are both human engineering and system engineering problems.

The Logicon-Extended CADSAT was developed in conjunction with the surveillance system requirements engineering activities. Since this work concentrated upon early requirements engineering (Type A and Type B5 specifications), the enhancements to the basic CADSAT satisfy a number of essential early requirements engineering needs. The extensions concentrate on the analyst needs for improved report generation. Liberties with the language were taken; the language features were employed differently in some cases than originally intended. The basic and the extended CADSAT satisfy most of the needs of the capabilities list and therefore the Guidebook. One approach would be to employ the basic CADSAT with certain extended CADSAT features, primarily to increase ease of use and provide additional reports. The second approach would be to add additional capabilities (reports) as well as to achieve the objectives of the first approach. These improvements would build upon the present design of CADSAT including its extensions.

7.   RESULTS AND RECOMMENDATIONS

The Requirements Engineering Guidebook

The Requirements Engineering Guidebook has been developed from analysis of past and present DoD and Air Force system engineering practices. It incorporates established requirements engineering techniques and approaches of many leading defense contractors. The Requirements Engineering Guidebook provides the necessary guidance for requirements engineering which is not described in current Air Force regulations, standards, or specifications. The Guidebook provides a general roadmap for performing

33

system requirements definition and analysis. It begins with a definition of the initial user requirements and continues through the complete definition and analysis of the system prior to its development. The Guidebook allows for a flexible approach in its application while providing the necessary guidance for government and contractor system analysts to plan and perform requirements engineering activities. It is recommended that the Guidebook be applied to selected programs to allow for clarification and improvement of its contents and presentation. The application of the Guidebook as a general guide may lead to a more formalized approach such as direct contract applications or a formal military standard. The relationship of the Requirements Engineering Guidebook to later phases of the Air Force system life cycle (such as in the full-scale development phase) should be studied and presented as an extension to the Requirements Standards Study.

CADSAT Enhancements

CADSAT has been found to be an effective tool for accomplishing the requirements engineering activities described in the Requirements Engineering Guidebook. Certain modifications, additions, and improvements to CADSAT have been identified during this study. These enhancements are oriented to improving the human engineering and system engineering process. The recommended improvements include simplifying the language, streamlining the analyzer to eliminate unnecessary reports, improving existing report capabilities, and increasing the overall performance and design of CADSAT. These enhancements would increase the effectiveness of CADSAT in support of the application of the Requirements Engineering Guidebook and would improve CADSAT's efficiency. Extensive use of CADSAT in the Air Force acquisition environment will require continuing enhancements to satisfy additional needs as described below.

Extended CADSAT Capabilities

Four promising uses of a requirements engineering tool like the Logicon-Extended CADSAT are (1) automated specification generation from the

requirements data base, (2) management information system applications, (3) additional query-reporting capabilities, and (4) simulation capabilities. The first three can be presently achieved to a limited degree by using CADSAT. Simulation using a requirements engineering tool is considered too experimental to recommend as an essential capability for a requirements engineering tool at this time. Improvements in automated specification generation from a requirements data base is considered the most beneficial enhancement to CADSAT. Extended query-reporting capabilities and management information system features would be the next most beneficial extension beyond the essential capabilities identified to support the Guidebook. Finally, simulation should continue to be investigated and experimental approaches encouraged. A thorough analysis of the benefits of simulation in the Air Force acquisition environment, the identification of requirements engineering simulation capabilities, and the development of the specified capabilities into current requirement engineering tools such as CADSAT is recommended as a principle area of research at this time.

Evolutionary Approach

The application of the Requirements Engineering Guidebook and the recommended changes to existing practices, regulations, standards, and specifications must proceed in a careful and selective manner. The incompatibility of the Guidebook with some current acquisition practices will demand changes to existing regulations, standards, and specifications over a period of time to allow the application of the Guidebook to evolve. Essential to the promotion of the Guidebook is adequate training for the engineers who must apply the Guidebook to their programs. The key element of this training will be to present the Guidebook as guidelines and standards for improved requirements engineering. The success of the documentation and analysis requirements of the Guidebook will depend upon the availability of requirements engineering tools like CADSAT. The intermediate documentation and analysis needs which are essential to the requirement engineering process are not considered to be easily accomplished without automated assistance. In addition, the specific

35

issues of the acquisition environment, the application of the Guidebook, and the use of automated tools must be addressed in specific methodologies for each acquisition environment as described below.

Requirements Engineering Methodology

The Requirements Engineering Guidebook presented in this report provides the procedural framework for the definition and analysis of system requirements for any Air Force systems development. The associated list of automated tool capabilities was developed to complement the Guidebook and to facilitate the definition, analysis, and documentation of the system requirements. Specific approaches for the application of the Guidebook within various acquisition environments and the integration of automated tool capabilities in support of the requirements engineering tasks described in the Guidebook can be facilitated by specific guidance - a requirements engineering methodology. The methodology provides the means of adapting the procedures and tools to specific acquisition environments and facilitates the introduction of the Guidebook and automated tools. The development of a requirements engineering methodology for the ESD acquisition environment based upon the Requirements Engineering Guidebook and automated assistance from CADSAT can proceed as an extension to the Requirements Standards Study. Additional guidelines for other acquisition environments can proceed based on the ESD methodology.

REFERENCES

[1]   AFSCM 375-5, Systems Engineering Management Procedures, 10 March
        1966, rescinded.

[2]   AFSCM 375-1 and revisions 375-1A, Configuration Management During
        Definition and Acquisition Phases, 1 June 1964, rescinded.

[3]   MIL-STD-490, Specification Practices, 30 October 1968.

[4]   MIL-STD-483 (USAF), Configuration Management Practices of Systems,
        Equipment, Munitions, and Computer Programs, 12 April 1971.

[5]   AFR 800-3, Engineering for Defense Systems, 30 August 1973.

[6]   MIL-STD-499A (USAF), Engineering Management, 1 May 1974.

[7]   L.A. Johnson, P.B. Merrithew, and D.G. Smith, "Automated Analysis
        of System Specifications, "The Second U.S. Army Software Symposium
        Logicon, Inc., Williamsburg, VA, 26-27 October 1978 pp. 235-258.

[8]   User Requirements Language (URL) User's Manual, Part I and II,
        H6180/Multics/Version 3.3, July 1978.

[9]   User Requirement Analyzer (URA) User's Manual, H6180/Multics/Version
        3.3, July 1978.

[10]   D. Teichroew, and E.A. Hershey III, "PSL/PSA: A Computer-Aided
        Technique for Structured Documentation and Analysis of Information
        Processing Systems," IEEE Trans. Software Eng., vol. SE-3 pp.
        41-48, January 1977.

[11]   Standard 7935.1-S, Automated Data Systems Documentation Standards,
        Office of the Assistant Secretary of Defense, 13 September 1977.

[12]   HIPO - "A Design Aid and Documentation Technique," IBM Manual,
        1974.

MISSION
of
Rome Air Development Center

RADC plans and executes research, development, test and selected acquisition programs in support of Command, Control Communications and Intelligence (C^3I) activities. Technical and engineering support within areas of technical competence is provided to ESD Program Offices (POs) and other ESD elements. The principal technical mission areas are communications, electromagnetic guidance and control, surveillance of ground and aerospace objects, intelligence data collection and handling, information system technology, ionospheric propagation, solid state sciences, microwave physics and electronic reliability, maintainability and compatibility.